

PEQUEÑA UNIDAD ARITMÉTICO-LÓGICA

1. ESPECIFICACIONES

Sobre la EMP7128SLC84-6, se quiere diseñar un circuito secuencial síncrono, que sirva como pequeña unidad aritmético-lógica, con 4 ciclos de operación:

1. Espera (Estado S0 del controlador CtrlUnit).
2. Carga1 del Dato A y Operación OP a efectuar (estado S1).
3. Carga2 del Dato B o Recirculación del Resultado R anterior (estado S2).
4. Operación/Validación del nuevo Resultado R (estado S3).

El sistema será síncrono, sensible a flanco de subida del reloj CLK, e incluirá un RST asíncrono y estará formado por dos bloques, tal como se puede apreciar en la figura 1.

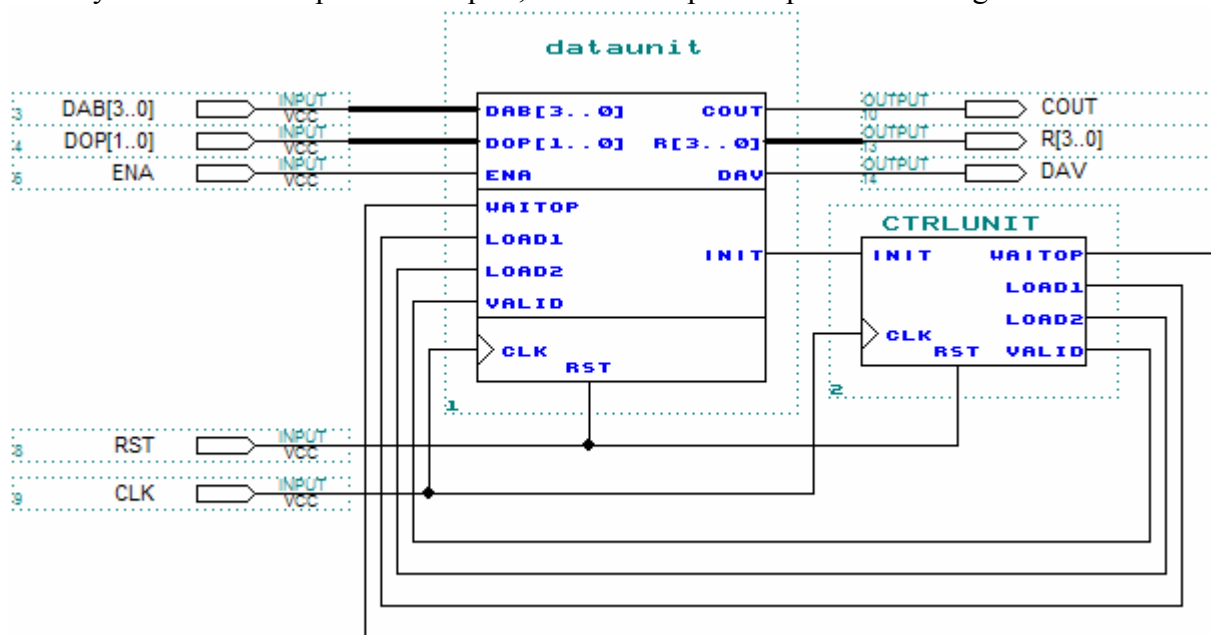


Figura 1. Diagrama de bloques de la Unidad Aritmético-Lógica.

El bloque Datos o DataUnit, incluirá la lógica necesaria para, bajo el control de CtrlUnit, realizar las siguientes operaciones:

1. En espera y sólo en espera (señal WaitOp proveniente de CtrlUnit), cuando la señal ENA (normalmente a '0') dé lugar a un pulso '1' de un ciclo de CLK de duración, DataUnit generará un pulso Init a '1' síncrono que indicará a CtrlUnit que debe iniciarse una nueva operación.
2. Al siguiente ciclo de CLK se capturarán el operando A, desde DAB[3..0], y la operación OP, desde DOP[1..0], a efectuar posteriormente mediante la señal Load1 proveniente de CtrlUnit.
3. Al siguiente ciclo de CLK se capturará el operando B desde DAB[3..0] o recirculará el resultado registrado anterior R[3..0]. Esta operación será síncrono y controlada por Load2 proveniente de CtrlUnit y la operación OP capturada en DOP[1..0] el ciclo anterior.
4. Las operaciones OP se efectuarán en una `lpm_add_sub`. CtrlUnit las comandará a través de la señal Valid y serán las de la Tabla 1, con A y B números binarios sin signo (supondremos que $R \geq 0$ siempre). No interesa el acarreo de entrada, pero sí el de Salida COUT. En el ciclo de operación/validación de datos DataUnit indicará mediante la señal DAV (activa a '0' en este ciclo de reloj sólo) que el contenido del registro R[3..0] y la señal combinacional COU son los

nuevos resultados válidos tras operación o RST. R[3..0] y la señal combinacional COUT son los nuevos resultados válidos tras operación o RST. R[3..0] habrá de permanecer hasta que cambie por otro resultado nuevo.

OP	Operación	Comentarios
00	A+B→R	Carga externa del dato B
01	A-B→R	Carga externa del dato B
10	A+R→R	Recirculación del dato anterior
11	A-R→R	Recirculación del dato anterior

Tabla 1. Operaciones de la Unidad Aritmético Lógica

El bloque Controlador o CtrlUnit, actúa como Control-Path habilitando el funcionamiento del bloque DataUnit en sus 4 ciclos de operación síncronamente. Debe advertirse que, una vez iniciada una operación, un nuevo pulso de Init devolverá el sistema a espera. El cronograma de la figura 2 describe el funcionamiento del sistema.

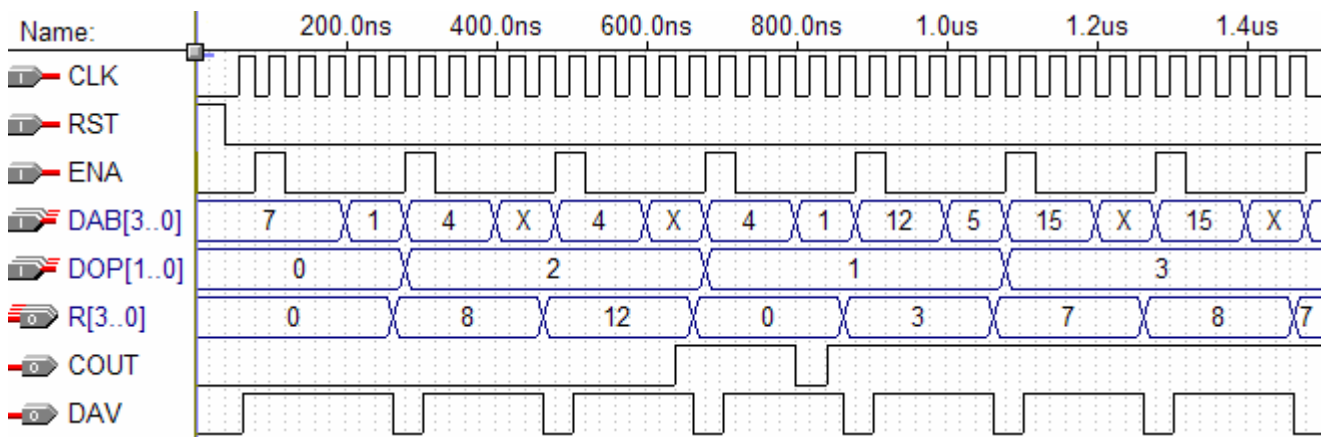


Figura 2. Cronograma de funcionamiento del sistema.

2. DESCRIPCIÓN DEL FUNCIONAMIENTO

El funcionamiento de la ALU será gobernado por una máquina de estados tipo Moore de 4

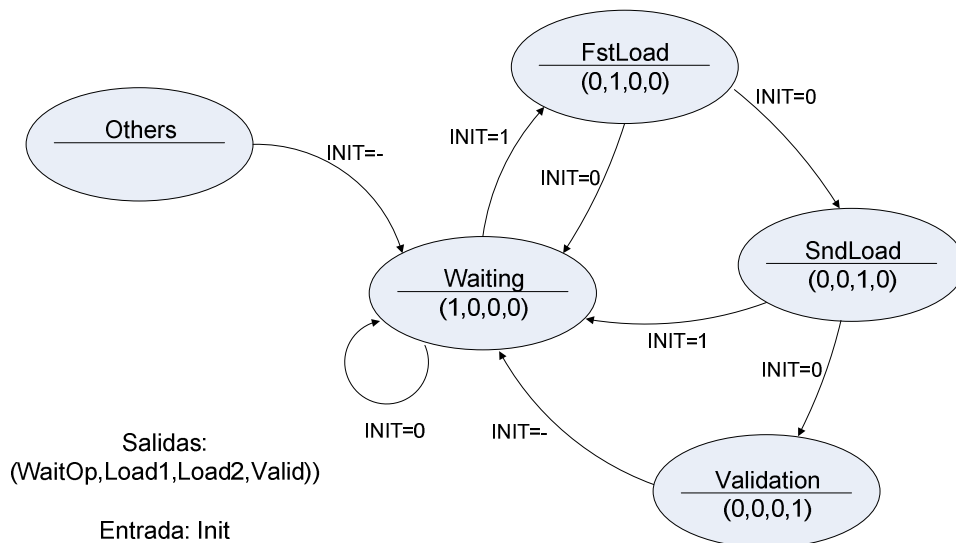


Figura 3. Máquina de estados del bloque CtrlUnit.

estados. Con este número de estados, sólo nos harían falta 2 registros para poder diseñarla, pero en este caso se aconseja diseñar una máquina de Moore cuyas salidas (WaitOp, Load1, Load2 y Valid) coincidan con los registros de estado (máquina de Medvedev). Este tipo de máquinas son las ideales en microelectrónica y carece de sentido codificar al máximo los estados (para ahorrar registros) si después debemos emplear otras LCs para decodificar salidas: lo mejor es optimizar el conjunto de registros de estados y salidas, minimizando el conjunto de LCs ocupadas.

3. ORGANIZACIÓN DE TAREAS

1. Diseño del bloque DataUnit
 - a. Implementar y compilar desde Captura de Esquemas
 - b. Verificar el bloque DataUnit. En la figura 4 se proporciona un esquemático del bloque DataUnit en el que se deberán editar nombres, parámetros y puertos, para que el diseño funcione correctamente
2. Diseño del bloque CtrlUnit
 - a. Implementar y compilar desde descripción VHDL
 - b. Verificar el bloque Control. En la figura 5 se proporciona un análisis temporal de cómo debería funcionar el bloque controlador.
3. Diseño de la jerarquía superior
 - a. Implementar y compilar desde captura de esquemas.
 - b. Verificar el bloque TOTAL
4. Rellenar
 - a. Número de LCs utilizadas
 - b. Fmax de operación
5. Realizar las siguientes asignaciones
 - a. RST como Global Clear
 - b. ENA como Entrada Dedicada

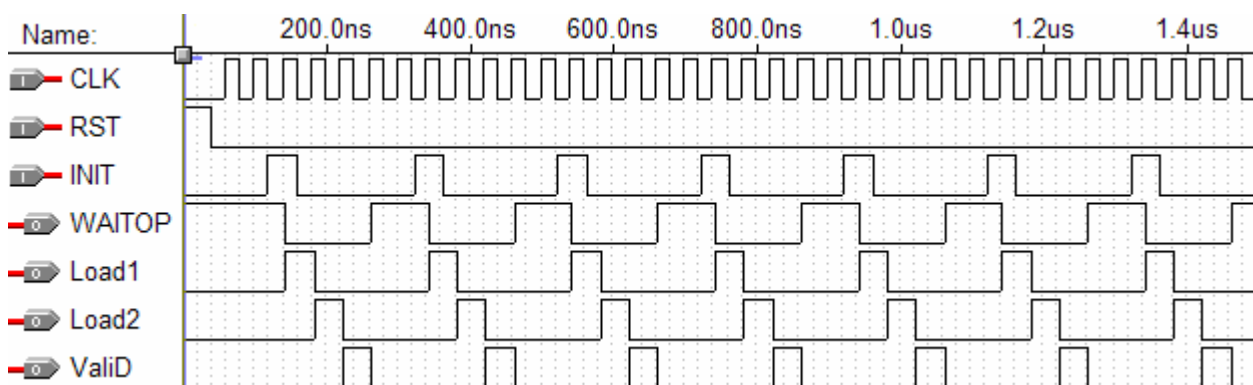


Figura 5. Verificación temporal del bloque CtrlUnit.

SISTEMAS ELECTRÓNICOS DIGITALES
4º DE INGENIERÍA SUPERIOR EN AUTOMÁTICA Y ELECTRÓNICA INDUSTRIAL

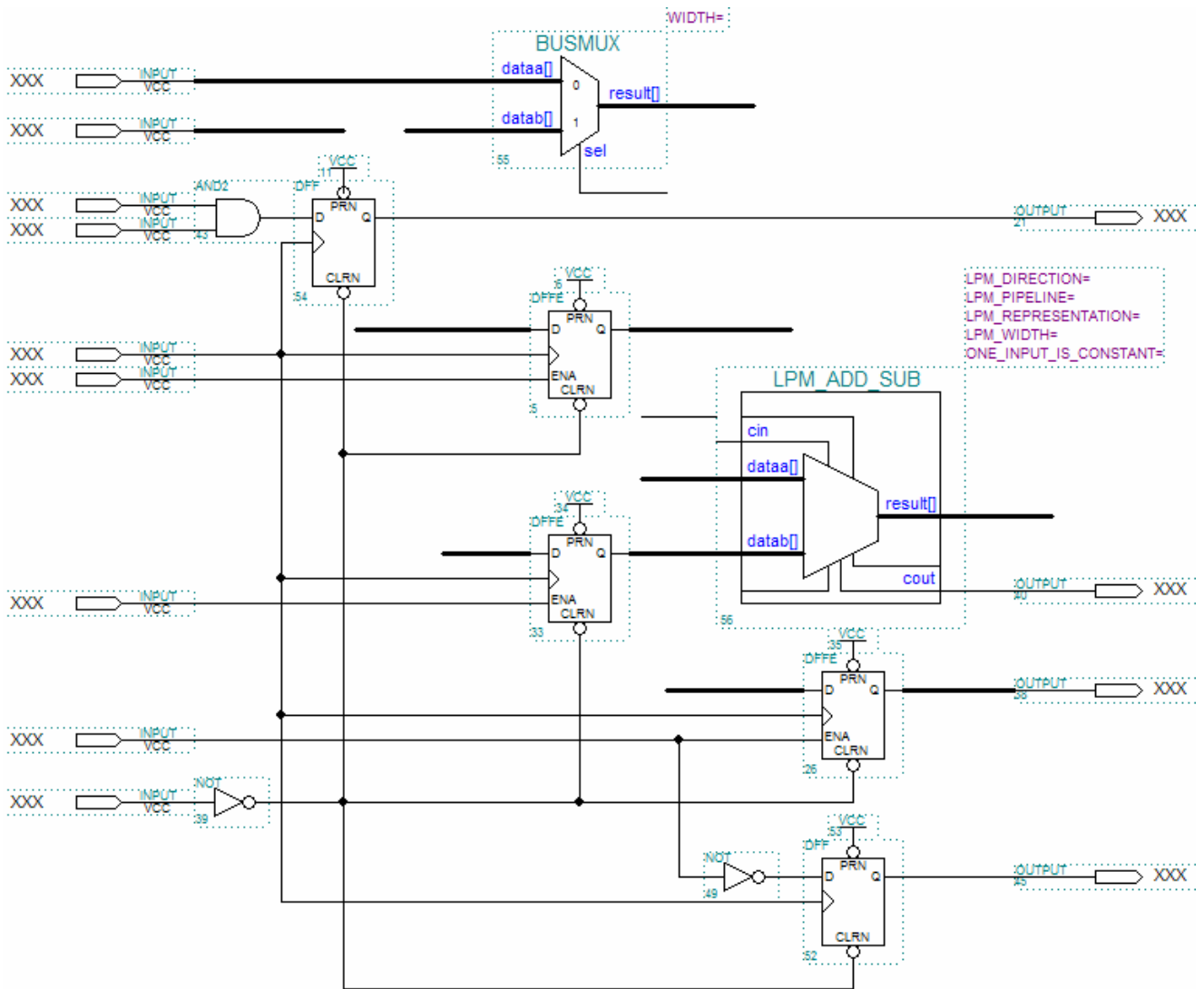


Figura 4. Esquemático incompleto del bloque DataUnit.